

Sage : un logiciel libre de mathématiques

Sébastien Labbé <slabqc@gmail.com>

LaCIM UQÀM

Cégep de Saint-Hyacinthe
19 janvier 2012

« Vous pouvez lire le théorème de Sylow et sa preuve dans le livre Huppert à la bibliothèque, puis vous pouvez alors utiliser le théorème de Sylow pour le reste de votre vie gratuitement. Mais pour plusieurs systèmes de calcul formel des redevances de licence doivent être versés régulièrement.

*Dans cette situation deux des règles les plus élémentaires de conduite en mathématiques sont violés. En mathématiques, les résultats sont **transmis gratuitement** et les démonstrations sont **publiques** pour permettre leur vérification. »*

—J. Neubüser (1993)
(il a fondé GAP en 1986)

« Je pense que, fondamentalement, le logiciel libre tend à être un logiciel plus stable. C'est la bonne façon de faire les choses. Je le compare à la *science versus la sorcellerie*.

En science, on se base sur les résultats des autres pour découvrir ou démontrer des résultats nouveaux.

Dans la sorcellerie, quelqu'un avait un petit secret bien gardé et jamais il ne permettait aux autres de vraiment le comprendre ni l'utiliser. » —Linus Torvalds

Why You Do Not Usually Need to Know about Internals

« You should realize at the outset that while knowing about the internals of Mathematica may be of intellectual interest, it is usually *much less important* in practice than you might at first suppose. [...] For the internals of Mathematica are *quite complicated*, [...], it is usually *extremely difficult* to reach a reliable conclusion about how the detailed implementation of this algorithm will actually behave in particular circumstances. »

—Mathematica Tutorial

<http://reference.wolfram.com/mathematica/tutorial/>

[WhyYouDoNotUsuallyNeedToKnowAboutInternals.html](http://reference.wolfram.com/mathematica/tutorial/WhyYouDoNotUsuallyNeedToKnowAboutInternals.html)

Plan

- 1 Début du logiciel Sage
- 2 Sage
- 3 Communauté
- 4 Quelques fonctionnalités

Plan

- 1 Début du logiciel Sage
- 2 Sage
- 3 Communauté
- 4 Quelques fonctionnalités

Début du logiciel Sage

Logiciels commerciaux existants :

- Maple, Waterloo Maple Inc., Maplesoft, depuis 1985.
- Mathematica, Wolfram Research, depuis 1988.
- Magma, University of Sydney, depuis 1990
- Matlab, MathWorks, depuis 1989

Logiciels libres existants :

- Maxima, William Schelter et coll., depuis 1967
- GAP, GAP Group, depuis 1986
- Singular, University of Kaiserslautern, depuis 1984
- PARI/GP, depuis 1985

Début du logiciel Sage

- *1999-2005.* William Stein écrit plus de 25000 lignes de code en Magma pour ses recherches mais les défauts de Magma le dérangent :
 - Impossibilité de définir ses propres objets,
 - pas de gestion des exceptions,
 - impossible d'écrire du code compilé,
 - le code de Magma est fermé,
 - Magma n'est pas gratuit.

« Je me suis assis sur un banc de parc et j'ai réalisé quel chemin dangereux j'étais en train de prendre en abandonnant autant de libertés si tôt dans ma carrière. À ce moment-là, j'ai fait la résolution de ne pas le faire et j'ai commencé à réfléchir à ce qui allait devenir Sage. »
— William Stein

Mathematical Software and Me : A Very Personal Recollection, 2009

Début du logiciel Sage

- *Jan. 2005.* William Stein crée Sage.
- *Fév. 2005.* Sage version 0.1 : rassemble 5 logiciels : PARI, Maxima, Python, Singular et GAP.
- *Déc. 2005.* William reçoit un courriel du leader de Magma.

Début du logiciel Sage

Date: Mon, 19 Dec 2005 16:54:09 -0800
From: "John Cannon" <john@maths.usyd.edu.au>
Subject: Magma calculator
William,

This is to formally advise you that your permission to run a general-purpose calculator based on Magma ends on Dec 31, 2005. This was originally set up at your request so students in your courses at Harvard could have easy access to Magma.

Please confirm receipt of this letter.
Wishing you a happy Christmas,
John

Début du logiciel Sage

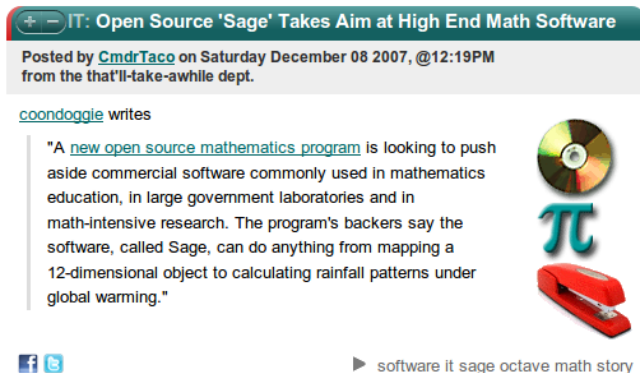
« Ce simple courriel m'a *sérieusement effrayé*. [...] Ce que John avait rendu clair comme du cristal était qu'il pouvait détruire tout mes plans à long terme sur la base d'un caprice. J'ai regardé autour de moi pour d'autres options et il n'y en avait pas. *Il fallait que Sage réussisse*. »

— William Stein

Mathematical Software and Me : A Very Personal Recollection, 2009

Début du logiciel Sage

- *Fév. 2006.* Sage version 1.0 est lancée ; et les premiers Jours Sage « annuels » sont organisés : Sage Days 1.
- *Déc. 2007.* Sage se fait remarquer par les nerds (slashdot.org) :




IT: Open Source 'Sage' Takes Aim at High End Math Software

Posted by [CmdrTaco](#) on Saturday December 08 2007, @12:19PM from the that'll-take-awhile dept.

[coondoggie](#) writes

"A [new open source mathematics program](#) is looking to push aside commercial software commonly used in mathematics education, in large government laboratories and in math-intensive research. The program's backers say the software, called Sage, can do anything from mapping a 12-dimensional object to calculating rainfall patterns under global warming."



► software it sage octave math story

Facebook Twitter

Plan

- 1 Début du logiciel Sage
- 2 Sage
- 3 Communauté
- 4 Quelques fonctionnalités

Mission

Sage est un logiciel libre de mathématiques dont la mission est d'offrir une alternative aux logiciels propriétaires Magma, Maple, Mathematica et Matlab et de favoriser le développement d'une communauté amicale d'utilisateurs et de développeurs.

Sage est un logiciel libre

Sage est distribué sous la version 2 de la **licence publique générale** (GPLv2) de GNU ce qui garantit quatre types de liberté :

- La liberté d'**utiliser** le programme (gratuitement).
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme à qui on veut**.

Sage est ...

une *distribution* de logiciels

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

ATLAS	Automatically Tuned Linear Algebra Software
BLAS	Basic Fortran 77 linear algebra routines
Bzip2	High-quality data compressor
Cddlib	Double Description Method of Motzkin
Common Lisp	Multi-paradigm and general-purpose programming lang.
CVXOPT	Convex optimization, linear programming, least squares
Cython	C-Extensions for Python
F2c	Converts Fortran 77 to C code
Flint	Fast Library for Number Theory
FpLLL	Euclidian lattice reduction
FreeType	A Free, High-Quality, and Portable Font Engine

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

G95	Open source Fortran 95 compiler
GAP	Groups, Algorithms, Programming
GD	Dynamic graphics generation tool
Genus2reduction	Curve data computation
Gfan	Gröbner fans and tropical varieties
Givaro	C++ library for arithmetic and algebra
GMP	GNU Multiple Precision Arithmetic Library
GMP-ECM	Elliptic Curve Method for Integer Factorization
GNU TLS	Secure networking
GSL	Gnu Scientific Library
JsMath	JavaScript implementation of LaTeX

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

IML	Integer Matrix Library
IPython	Interactive Python shell
LAPACK	Fortan 77 linear algebra library
Lcalc	L-functions calculator
Libgrypt	General purpose cryptographic library
Libgpg-error	Common error values for GnuPG components
Linbox	C++ linear algebra library
Matplotlib	Python plotting library
Maxima	computer algebra system
Mercurial	Revision control system
MoinMoin	Wiki

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

MPFI	Multiple Precision Floating-point Interval library
MPFR	C library for multiple-precision floating-point computations
ECLib	Cremona's Programs for Elliptic curves
NetworkX	Graph theory
NTL	Number theory C++ library
Numpy	Numerical linear algebra
OpenCDK	Open Crypto Development Kit
PALP	A Package for Analyzing Lattice Polytopes
PARI/GP	Number theory calculator
Pexpect	Pseudo-tty control for Python
PNG	Bitmap image support

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

PolyBoRi	Polynomials Over Boolean Rings
PyCrypto	Python Cryptography Toolkit
Python	Interpreted language
Qd	Quad-double/Double-double Computation Package
R	Statistical Computing
Readline	Line-editing
Rpy	Python interface to R
Scipy	Python library for scientific computation
Singular	fast commutative and noncommutative algebra
Scons	Software construction tool
SQLite	Relation database

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

Sympow	L-function calculator
Symmetrica	Representation theory
Sympy	Python library for symbolic computation
Tachyon	lightweight 3d ray tracer
Termcap	for writing portable text mode applications
Twisted	Python networking library
Weave	Tools for including C/C++ code within Python
Zlib	Data compression library
ZODB	Object-oriented database

Sage est une distribution de logiciels libres

Dans Sage, il y a les logiciels suivants

Sympow	L-function calculator
Symmetrica	Representation theory
Sympy	Python library for symbolic computation
Tachyon	lightweight 3d ray tracer
Termcap	for writing portable text mode applications
Twisted	Python networking library
Weave	Tools for including C/C++ code within Python
Zlib	Data compression library
ZODB	Object-oriented database

...et plus !


```
> sage -singular
```

```
                SINGULAR                               / Development
A Computer Algebra System for Polynomial Computations / version 3-1-0
                by: G.-M. Greuel, G. Pfister, H. Schoenemann 0<
FB Mathematik der Universitaet, D-67653 Kaiserslautern \ Mar 2009
> \
```

```
> sage -maxima
```

```
Maxima 5.16.3 http://maxima.sourceforge.net
```

```
Using Lisp ECL 9.4.1
```

```
Distributed under the GNU Public License. See the file COPYING.
```

```
Dedicated to the memory of William Schelter.
```

```
The function bug_report() provides bug reporting information.
```

```
(%i1)
```

```
> sage -gp
```

```
GP/PARI CALCULATOR Version 2.3.3 (released)
amd64 running linux (x86-64/GMP-4.2.1 kernel) 64-bit version
compiled: Jul 10 2009, gcc-4.3.2 (Ubuntu 4.3.2-1ubuntu12)
(readline v5.2 enabled, extended help available)
```

```
Copyright (C) 2000-2006 The PARI Group
```

```
PARI/GP is free software, covered by the GNU General Public License, and
comes WITHOUT ANY WARRANTY WHATSOEVER.
```

```
Type ? for help, \q to quit.
```

```
Type ?12 for how to get moral (and possibly technical) support.
```

```
parisize = 8000000, primelimit = 500000
```

```
?
```

```
> sage -R
```

```
R version 2.6.1 (2007-11-26)
```

```
Copyright (C) 2007 The R Foundation for Statistical Computing
```

```
ISBN 3-900051-07-0
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.
```

```
You are welcome to redistribute it under certain conditions.
```

```
Type 'license()' or 'licence()' for distribution details.
```

```
    Natural language support but running in an English locale
```

```
R is a collaborative project with many contributors.
```

```
Type 'contributors()' for more information and
```

```
'citation()' on how to cite R or R packages in publications.
```

```
Type 'demo()' for some demos, 'help()' for on-line help, or
```

```
'help.start()' for an HTML browser interface to help.
```

```
Type 'q()' to quit R.
```

```
>
```

Sage combine la puissance de
plusieurs logiciels.

Sage combine les logiciels

[Ces exemples sont tirés d'une présentation de William Stein]

Sage combine les logiciels

[Ces exemples sont tirés d'une présentation de William Stein]

Construction d'une courbe elliptique utilisant la table de *John Cremona* :

```
sage: E = EllipticCurve('389a')
```

Sage combine les logiciels

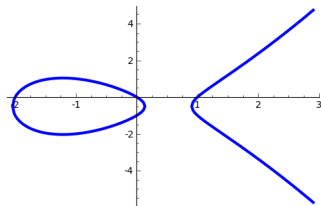
[Ces exemples sont tirés d'une présentation de William Stein]

Construction d'une courbe elliptique utilisant la table de *John Cremona* :

```
sage: E = EllipticCurve('389a')
```

Utilisation de *matplotlib* pour la dessi

```
sage: plot(E,thickness=3)
```



Sage combine les logiciels

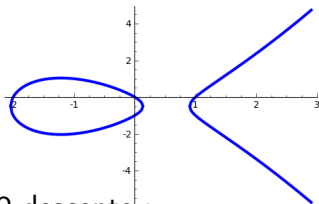
[Ces exemples sont tirés d'une présentation de William Stein]

Construction d'une courbe elliptique utilisant la table de *John Cremona* :

```
sage: E = EllipticCurve('389a')
```

Utilisation de *matplotlib* pour la dessi

```
sage: plot(E,thickness=3)
```



Utilisation de *mwrnk* pour faire une 2-descente :

```
sage: E.mwrnk()
```

```
Curve [0,1,1,-2,0] : Rank = 2
```

Sage combines software

Calcul des coefficients de Fourier a_n avec le logiciel *PARI* :

```
sage: E.anlist(15)
```

```
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6
```

Sage combines software

Calcul des coefficients de Fourier a_n avec le logiciel *PARI* :

```
sage: E.anlist(15)
```

```
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6]
```

Calculer les zéros dans la bande critique de la série L avec *Lcalc* :

```
sage: E.lseries().zeros(5)
```

```
[0.000000000, 0.000000000, 2.87609907, 4.41689608, 5.79
```

Sage combines software

Calcul des coefficients de Fourier a_n avec le logiciel *PARI* :

```
sage: E.anlist(15)
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6]
```

Calculer les zéros dans la bande critique de la série L avec *Lcalc* :

```
sage: E.lseries().zeros(5)
[0.000000000, 0.000000000, 2.87609907, 4.41689608, 5.79
```

Sympow pour calculer le degré modulaire :

```
sage: E.modular_degree()
40
```

Sage combines software

Calcul des coefficients de Fourier a_n avec le logiciel *PARI* :

```
sage: E.anlist(15)
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6]
```

Calculer les zéros dans la bande critique de la série L avec *Lcalc* :

```
sage: E.lseries().zeros(5)
[0.000000000, 0.000000000, 2.87609907, 4.41689608, 5.79
```

Sympow pour calculer le degré modulaire :

```
sage: E.modular_degree()
40
```

Calcul du rang du groupe 3-Selmer avec le logiciel (non libre)

Magma :

```
sage: magma(E).ThreeSelmerGroup()
Abelian Group isomorphic to Z/3 + Z/3
Defined on 2 generators
```

Sage combine les logiciels

```
sage: integrate(cos(x^2), x)
-1/8*((I + 1)*sqrt(2)*erf((1/2*I - 1/2)*sqrt(2)*x) +
(I - 1)*sqrt(2)*erf((1/2*I + 1/2)*sqrt(2)*x))*sqrt(pi)
```

Logiciels utilisés pour faire ce calcul :

```
sage: from sage.misc.citation import get_systems
sage: get_systems("integrate(cos(x^2), x)")
['MPFI', 'ginac', 'GMP', 'Maxima']
```

Sage combine les logiciels

« Nous mettons en oeuvre toutes les routines de conversion, au lieu de s'attendre à ce que les logiciels en amont le fasse. Nous les faisons communiquer avec Sage qu'ils le veulent ou non. La résistance est futile. »

*—William Stein
(fondateur de Sage)*

Le langage de programmation de Sage
est Python

Le langage de programmation de Sage Python

Python est un langage de programmation très puissant, moderne, et interprété.

- Facile à lire et à écrire :

math : $\{17x \mid x \in \{0, 1, \dots, 9\} \text{ et } x \text{ est impair}\}$
python : `[17*x for x in range(10) if x % 2 == 1]`

- Accès à plusieurs **librairies** de Python base de données, graphiques, réseau, . . .
- Utilisation des **librairies C/C++** à partir de Python.

Le langage de programmation de Sage est Python

« *Google has made no secret of the fact they use Python a lot for a number of internal projects. Even knowing that, once I was an employee, I was amazed at how much Python code there actually is in the Google source code system.* »

— Guido van Rossum

(créateur de Python)

Plan

- 1 Début du logiciel Sage
- 2 Sage
- 3 Communauté**
- 4 Quelques fonctionnalités

Carte des contributeurs



Il y a présentement $242 + 2$ contributeurs
de $164 + 1$ places différentes de partout dans le monde.

Les jours Sage

Les jours Sage sont des ateliers ayant pour objectifs de

- développer de nouvelles fonctionnalités
- attirer de nouveaux utilisateurs et développeurs

Les algorithmes innovateurs développés sont souvent soumis pour publications dans des journaux.

Plus de 30 ateliers ont été organisés à ce jour dans le monde.

Jours Sage en 2010

- Sage Days 19 : Seattle, USA (January 2010)
- Sage Days 20 : Marseille, France (February 2010)
- Sage Days 20.25 : Montreal, Canada (March 2010)
- Sage Days 20.5 : Toronto, Canada (May 2010)
- Sage Days 21 : Seattle, USA (June 2010)
- Sage-Combinat/Chevie : France (June 2010)
- Sage Days 22 : Berkeley, USA (July 2010)
- Sage Days 23 : Leiden, Netherlands (July 2010)
- Sage Days 23.5 : Kaiserslautern, Germany (July 2010)
- Sage Days 24 : Linz, Austria (July 2010)
- Sage Days 25 : Mumbai, India (August 2010)
- Sage Days 25.5 : Montreal, Canada (September 2010)
- Sage Days 26 : Seattle, USA (December 2010)

Jours Sage en 2011

- Sage Days 27 : Seattle, USA (January 2011)
- Sage Days 28 : Orsay, France (January 2011)
- Sage Days 29 : Seattle, USA (March 21-25, 2011)
- Sage Days 30 : Halifax, Canada (May 2-6, 2011)
- Sage Days 31 : Seattle, USA (June 11-18, 2011)
- Sage Education Days 3 : Seattle, USA (June 16-18, 2011)
- Sage Days 32 : Seattle, USA (August 21-25, 2011)
- Sage Days 33 : Women in Sage, Seattle (September 2011)
- Sage Days 34 : Kaiserslautern, Germany (Sept. 2011)
- Sage Days 34.5 : Bobo Dioulasso, Burkina Faso (Nov. 2011)
- Sage Days 35 : Warwick UK (December 2011)

Jours Sage en 2012

- Sage Days 35.5 : Wenham, MA (January 2012)
- Sage-Combinat Days 35.75 : Cernay-la-Ville, France (February 2012)
- Sage Days 36 : UC San Diego, (February 2012)
- Sage Days X, for some $X > 36$: South Korea (April 2012)
- Sage Days Y, for some $Y > X$: CRM, Montréal (Mai 2012)
- Sage Days M, for some $Y > M$: Seattle, USA (Juin 2012)
- Sage Days : February 11-15, 2013

Plan

- 1 Début du logiciel Sage
- 2 Sage
- 3 Communauté
- 4 Quelques fonctionnalités**

Quelques fonctionnalités

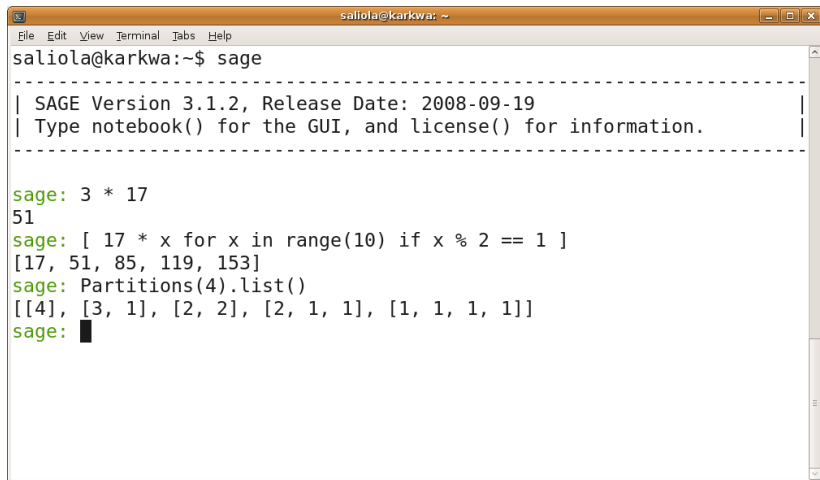
Interfaces :

- Ligne de commande
- Sage Notebook (en local)
- Sage Notebook (sur internet)
- **Nouveau** : Sage dans une page web (par Jason Grout, États-Unis)

Fonctionnalités intéressantes :

- Sagetex (Dan Drake, Corée du Sud)
- Graph Editor (Radoslav Kirov, États-Unis)
- Interact et animations
- Quantumino Solver, (L., Montréal)
- **Cython** : traduction du code Python \mapsto code C.

La ligne de commande



```
saliola@karkwa: ~  
File Edit View Terminal Tabs Help  
saliola@karkwa:~$ sage  
-----  
| SAGE Version 3.1.2, Release Date: 2008-09-19  
| Type notebook() for the GUI, and license() for information.  
-----  
  
sage: 3 * 17  
51  
sage: [ 17 * x for x in range(10) if x % 2 == 1 ]  
[17, 51, 85, 119, 153]  
sage: Partitions(4).list()  
[[4], [3, 1], [2, 2], [2, 1, 1], [1, 1, 1, 1]]  
sage: █
```

L'interface Notebook (en local)

The screenshot shows the Sage Notebook interface in a web browser. The browser address bar shows `localhost:8000/home/admin/3/`. The page title is "The Sage Notebook" with version 4.5.3. The navigation menu includes links for `admin`, `Toggle`, `Home`, `Published`, `Log`, `Settings`, `Help`, `Report a Problem`, and `Sign out`. The main content area shows a code editor with the following code:

```
plot(sin(x^2)+cos(x), -pi, pi, hue=0.8, thickness=4).show(figsize=[8,2])
```

Below the code editor is a plot of the function $y = \sin(x^2) + \cos(x)$ over the interval $x \in [-\pi, \pi]$. The plot shows a smooth, oscillating curve with a peak near $x = -1$ and a trough near $x = 1$. The axes are labeled with values from -3 to 3 on the x-axis and -1.5 to 1 on the y-axis.

Below the plot is a code editor window titled "plot" with the following content:

EXAMPLES: We plot the sin function:

```
sage: P = plot(sin, (0,10)); print P
Graphics object consisting of 1 graphics primitive
sage: len(P) # number of graphics primitives
1
sage: len(P[0]) # how many points were computed (random)
225
sage: P # render

sage: P = plot(sin, (0,10), plot_points=10); print P
Graphics object consisting of 1 graphics primitive
sage: len(P[0]) # random output
32
sage: P # render
```

Below the code editor window, there is a text box with the following text:

We plot with `randomize=False`, which makes the initial sample points evenly spaced (hence always the same). Adaptive plotting might insert other points, however, unless

L'interface Notebook (sur internet)

`http://sagenb.org/`
`http://sage.lacim.uqam.ca/`

 The Sage Notebook
Version 4.7

Welcome!

Sage is a different approach to mathematics software.

The Sage Notebook

With the Sage Notebook anyone can create, collaborate on, and publish interactive worksheets. In a worksheet, one can write code using Sage, Python, and other software included in Sage.

General and Advanced Pure and Applied Mathematics

Use Sage for studying calculus, elementary to very advanced number theory, cryptography, commutative algebra, group theory, graph theory, numerical and exact linear algebra, and more.

Use an Open Source Alternative

By using Sage you help to support a viable open source alternative to Magma, Maple, Mathematica, and MATLAB. Sage includes many high-quality open source math packages.

Use Most Mathematics Software from Within Sage

Sage makes it easy for you to use most mathematics software together. Sage includes GAP, GP/PARI, Maxima, and Singular, and dozens of other open packages.

Use a Mainstream Programming Language

You work with Sage using the highly regarded scripting language Python. You can write programs that combine serious mathematics with anything else.

Sign into the Sage Notebook v4.7

Username

Password

Remember me

[Sign up for a new Sage Notebook account](#)

[Browse published Sage worksheets](#)
(no login required)

Sage dans une page web

<http://www.thales.math.uqam.ca/~labbes/blogue/>

Sébastien Labbé

[Blogue \(index\)](#) [Publications](#) [Communications](#) [Sage](#) [Ultimate](#)

Utiliser Sage dans une page web

14 octobre 2011 | [Catégorie: sage](#) | [0 Commentaire](#) et [0 Réaction](#)

Lorsque j'ai participé aux Sage Days 31 à Seattle en juin dernier (conférence dont l'objectif était de développer le Sage Notebook), j'ai rencontré Jason Grant (Drake University, Iowa). Le projet de Jason était de parvenir à utiliser Sage directement dans une page web (voir la discussion [public single cell](#) sur sage-devel pour en savoir plus).

Son projet est toujours en cours, mais il semble déjà être fonctionnel à en juger la page web de Rob Beezer (University of Puget Sound, Washington) qui illustre un exemple en faisant des calculs sur le graphe de Petersen et le calcul d'un factoriel.

Essays d'imiter ce qu'il a fait! Dans la cellule qui suit, on définit une fonction de deux variables $f(x,y) = \frac{-xy}{x^2+y^2}$ et on affiche ses lignes de niveau. Cliquez sur le bouton `Calculer` pour les afficher. Vous pouvez aussi changer la fonction et réévaluer la cellule.

```

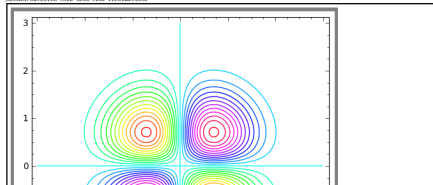
1: f(x,y) = x * y / exp(x^2 + y^2)
2: d = implicit_plot(f(x,y), (x,-2,2), (y,-2,2), contours=30, cmap='hsv')
3: d.show(aspect_ratio=1)

```

Calculer

Session 8af98166-93eb-4b38-9f8b-41561ab9a03f

Toggle Plaintext Input



\LaTeX

Dans ce fichier \LaTeX , j'écris :

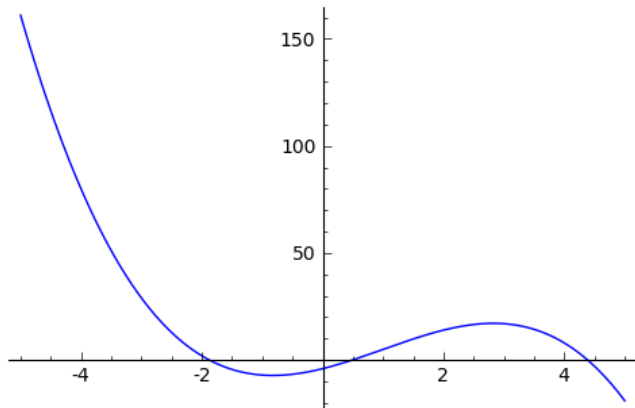
```
\sageplot{plot(-x^3+3*x^2+7*x-4,-5,5)}
```

\LaTeX

Dans ce fichier \LaTeX , j'écris :

```
\sageplot{plot(-x^3+3*x^2+7*x-4,-5,5)}
```

et c'est remplacé par :



L^AT_EX

Dans ce fichier L^AT_EX :

```
\begin{sagesilent}
  sigma = Permutation([7,3,1,5,2,6,8,4])
  P, Q = sigma.robinson_schensted()
\end{sagesilent}
```

Soit la permutation $\sigma = \text{sage}\{\sigma\}$.
L'algorithme de Robinson-Schensted-Knuth produit
les tableaux:

```
\[\text{sage}\{P\} \quad \text{sage}\{Q\}\]
```

L^AT_EX

Cela a été remplacé par :

Soit la permutation $\sigma = [7, 3, 1, 5, 2, 6, 8, 4]$.
L'algorithme de Robinson-Schensted-Knuth produit les tableaux :

1	2	4	8
3	5	6	
7			

1	4	6	7
2	5	8	
3			

L^AT_EX

Cela a été remplacé par :

Soit la permutation $\sigma = [7, 3, 1, 5, 2, 6, 8, 4]$.
L'algorithme de Robinson-Schensted-Knuth produit les tableaux :

1	2	4	8
3	5	6	
7			

1	4	6	7
2	5	8	
3			

Ceci est rendu possible par le paquetage *sagetex* écrit par Dan Drake. Bien sûr, il est inclu dans Sage.

Pour plus d'informations

<http://sagemath.org>

The screenshot shows the SageMath website homepage. At the top left is the Sage logo (a dodecahedron) and the text 'sage'. To the right, it displays 'v4.7.1 (2011-08-11)' and social media icons for Facebook, Twitter, and LinkedIn, along with a '177' count. Below this is a search bar and a navigation menu with links: Home, Tour, Support, Library, Download, Development, Links.

The main content area features a paragraph: "Sage is a free open-source mathematics software system licensed under the GPL. It combines the power of many existing open-source packages into a common Python-based interface. Mission: Creating a viable free open source alternative to Magma, Maple, Mathematica and Matlab."

Below this are six main navigation buttons arranged in a 2x3 grid:

- Try Sage Online**: Includes sub-links for 'other: KAIST' and 'testing: alpha, Solaris SPARC'.
- Download 4.7.1**: Includes sub-links for 'Changelog', 'Source 4.7.1', and 'Packages'.
- Help/Documentation**: Includes sub-links for 'Video', 'Lists', 'Tutorial', 'FAQ', and 'Ask'.
- Feature Tour**: Includes sub-links for 'Quickstart', 'Research', and 'Graphics'.
- Library**: Includes sub-links for 'Testimonials', 'Books', 'Publications', and 'Press Kit'.
- Search**: Includes a search input field.

At the bottom, there is a 'Random Link: Publications citing Sage' and a vertical 'Follow us' button on the right side.

Pour plus d'informations

- <http://sagemath.org>
- <http://sagemath.org/doc>
- <http://sagemath.org/help.html>
- <http://sagemath.org/download.html>
- <http://sagemath.org/fr>
- <http://sagemath.org/fr/html/tutorial>
- <http://wiki.sagemath.org>

Le livre **Calcul mathématique avec Sage** :

- <http://sagebook.gforge.inria.fr/>

S. Labbé, **Sage, un logiciel libre de mathématiques**, Bulletin de l'AMQ, Vol. 51, no 2, mai 2011, p. 59–67.

Démo

Exercices

- Créer un compte sur <http://sagenb.org> ou <http://sage.lacim.uqam.ca/>
- Faîtes un calcul de votre choix
- Trouver et tester un exemple de votre choix dans le tutorial en français
<http://sagemath.org/fr/html/tutorial>
- Trouver et tester un exemple de votre choix dans le wiki de Sage <http://wiki.sagemath.org>
- Importer (upload) les feuilles de travail Sage sur mon site web
<http://www.thales.math.uqam.ca/~labbes/Sage/>